HIERARCHICAL FORECASTING OF WEB SERVER WORKLOAD USING SEQUENTIAL MONTE CARLO TRAINING

Tom Vercauteren, Pradeep Aggarwal†, Xiaodong Wang†, Ta-Hsin Li‡*

* INRIA, Sophia Antipolis, France,
† Department of Electrical Engineering, Columbia University, New York, NY 10027
‡ department of Mathematical Sciences, IBM T.J. Watson Research Center, Yorktown Heights, NY 10598

ABSTRACT

We propose a solution to the web server load prediction problem based on a hierarchical framework with multiple time scales. This framework leads to adaptive procedures that provide both long-term (in days) and short-term (in minutes) predictions with simultaneous confidence bands which accommodate not only serial correlation but also heavy-tailedness, and non-stationarity of the data. The long-term load is modeled as a dynamic harmonic regression (DHR), the coefficients of which evolve according to a random walk, and are tracked using sequential Monte Carlo (SMC) algorithms; whereas, the short-term load is predicted using an autoregressive model, whose parameters are also estimated using SMC techniques. We evaluate our method using real world web workload data.

1. INTRODUCTION

A web server farm is a cluster of servers shared by several web applications and services, and maintained by a host service provider. Usually the owner of the web applications pays the host service provider for the computing resources, and in return gets a quality-of-service (QoS) guarantee, which promises a certain minimum level of resources and performance. Static allocation of resources at the server farm is not efficient, therefore, the server farm allocates the computing resources dynamically among the competing applications to meet the quality-of-service for different classes of service requests. The requirement for dynamic allocation of resources makes it necessary for the server farm to be able to predict the workload accurately, with a sufficiently long time horizon to ensure that adequate resources are allocated to the services in-need in a *timely* manner [1, 2].

The server workload is usually measured in terms of the amount of services request per unit time. A time series of such a workload is quite challenging to predict accurately. In particular, the bursty nature and the non-stationarity of the server workload impose inherent limits on the accuracy of the prediction. Such a time series can, for example, be stationary but self-similar, and/or heavy-tailed over small duration (seconds or minutes) at a fine time granularity [3, 4]; it can also exhibit strong daily and weekly patterns (seasonality), which change randomly over different times of the day and different days of the week, and can also show calendar effects (different patterns on weekends) [4]. It is this second type of data with seasonal variations which is key to the designing of dynamic resource allocation schemes, and is the focus of the current paper.

The traditional linear-regression-based methods can give predictions with a limited accuracy, since the model can become inefficient in the presence of correlated error. In this paper, we follow the hierarchical approach proposed in [5, 6] in which the time series prediction is decomposed into two steps: first a prediction of the long-term component, which primarily captures the non-stationarity of the data, is performed and then the residual short-term process, which captures both the long-term prediction error and the short-term component of the time series, is processed. In this work, the long-term component is modeled as a linear combination of certain basis functions with random amplitudes evolving with time, while the residual short-term process is modeled as a traditional AR process. The parameters for both the short-term model and the long-term model are estimated using sequential Monte Carlo (SMC) methods (see [7, 8] and the references therein). The proposed method, in addition to providing predictions, can also be used to compute confidence bands simultaneously. This is of major interest in this setting since quantiles, as opposed to a simple prediction of the time series, can be used to support flexible (probability based) service-level agreements. Further, the proposed model allows the model parameters to change with time, thereby making itself capable of handling the non-stationarity in the data.

2. THE HIERARCHICAL FRAMEWORK

We consider a typical web-server farm, which records the number of requests at each server and aggregates them over small time intervals of length $\Delta > 0$ to obtain a time series. The data is non-stationary in that the mean changes with

time-of-day and day-of-week. It is also observed that the time series shows predominant daily patterns, varying randomly. Let p denote the sampling frequency for the daily pattern. For the example shown in Fig. 1, for $\Delta = 5$ minutes, $p = 288$. Although, several methods exist for modeling such a time series, we follow the hierarchical approach developed in [6]. Fig. 1 shows the hierarchical structure of the time-series, where the data is decomposed into a periodic long-term component and a randomly fluctuating short-term component.

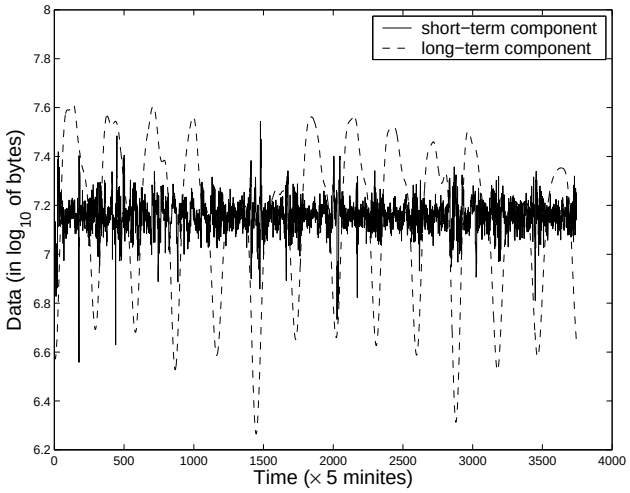

Fig. 1. The decomposition of the web-server data into a long-term pattern and a short-term random components.

Let $y(t)$ denote the observed load (after taking logarithm) at a server at time t. In order to capture the seasonality in the data, we use the dynamic harmonic regression (DHR) model of [9]. Stochastic time-varying parameters are used to characterize the various components of the DHR thus allowing for non-stationarity in the resulting time series. In practice, not all components of the DHR are necessary and in this paper we focus on the seasonal component. In our hierarchical framework, the time series is first modeled as a combination of a periodic long-term pattern $p(t)$, and a more irregular short-term component $e(t)$:

$$y(t) = p(t) + e(t). \tag{1}$$

The periodic long-term pattern $p(t)$ is represented as a weighted sum of some p-periodic basis functions $\phi_j(t)$ as

$$p(t) = \sum_{j \in \mathcal{P}_t} a_j(t)\phi_j(t), \tag{2}$$

where $a_j(t)$ are stochastic time-varying parameters and $\{\phi_j, j \in \mathcal{P}_t\}$ forms a subset of some linearly independent basis function $\{\phi_j, j \in \mathcal{P}\}$. To filter out the long-term component in (1), we choose the periodic basis functions to be sinusoidal waves whose frequencies are chosen based on the spectral properties of the time series (more on this in Section 3.1.1), giving us a harmonic regression (HR) on the long-term component.

Once the long-term estimation $\hat{y}_L$ has been performed, our hierarchical framework focuses on making an accurate d-step forecast of the residual time series,

$$z(t) = y(t) - \hat{y}_L(t|t). \tag{3}$$

This d-step forecast process is modeled as an AR process,

$$z(t+d) = \sum_{i=1}^{q_t} b_i(t) \cdot z(t-i+1) + \varepsilon(t), \tag{4}$$

the parameters of which (order, coefficients, and noise characteristics) being stochastic time-varying parameters as in [10]. Furthermore, in order to accommodate for the shot noises in the data, we use heavy-tailed distributions for the noise term $\varepsilon(t)$. The observation models (1), and (4), together with their dynamically varying coefficients form two dynamic state-spaces, both are tracked using the SMC methods.

For the time series of service requests, it is typical to have weekday patterns behaving significantly differently from the weekend patterns. In this paper, a multiple regime approach is employed, in which data belonging to different regimes are modeled separately to take advantage of the within-regime resemblance. The data belonging to the same regime is cascaded to obtain a set of new time series, one for each regime. Each time series is then modeled by (2) and (4), each regime having its own set of parameters.

3. LONG-TERM MODELING AND PREDICTION

Selection of the basis set $\mathcal{P}$ reduces the dimensionality of the problem, hence reducing the computational complexity as well as storage requirement associated with the training, modeling and prediction. Indeed, the higher is the dimension, the greater is the number of parameters to be estimated. We will see in Section 3.1.1 that only the first few frequencies (including the zero-frequency term) affect the seasonal variations to any significant extent, and that only a fixed number of sinusoids need to be in $\mathcal{P}$. It turns out that usually even within this fixed subset, only a few among those chosen frequency components are significant in representing the model at a particular time t, while the remaining ones can be discarded without significant loss in performance. However, the important subset of $\mathcal{P}$ can change with time. Therefore, instead of accommodating all of them in our model, we can reduce the dimensionality further by dynamically selecting the frequencies from the set $\mathcal{P}$, as the system evolves. We follow the jump Markov framework of [10], which is close to the resampling-based shrinkage

method, proposed in [11] in the context of blind detection in fading channels.

Let us now write down the state-space form we use for the long-term model:

$$\begin{aligned} y(t) &= \sum_{j\in\mathcal{P}_t} a_j(t)\phi_j(t) + e(t), \\ a_j(t) &= a_j(t-1) + v_j(t), \quad \forall j \in \mathcal{P}, \end{aligned} \tag{5}$$

where $\boldsymbol{v}(t) = \{v_j(t), j \in \mathcal{P}\}$ is the process noise and $e(t)$ is the measurement noise, and $y(t)$ is the observed data. The second equation in (5) represents the first-order Markov transition process, which is assumed to generate the coefficients vector $\boldsymbol{a}(t) = \{a_j(t), j \in \mathcal{P}\}$. The vector $\boldsymbol{v}(t)$ represents temporally uncorrelated Gaussian disturbances with zero mean, and covariance matrix $\boldsymbol{Q}_v$. The zero-mean noise in the state equation stems from the assumption that the long-term behavior is periodic with slow variations, for which, the incremental mean of the coefficients is close to zero. The first equation in (5) represents the measurement equation, where $e(t)$ is the temporally uncorrelated Gaussian disturbance with zero mean and variance σ_e^2. The initial state vector $\boldsymbol{a}(0)$ is assumed to be Gaussian distributed with mean $\bar{\boldsymbol{a}}(0)$ and covariance matrix $\boldsymbol{P}(0)$, which are computed as the respective mean and covariance of the coefficients of the harmonics included in the regression, obtained from the training data.

3.1. Determination of the Fixed Parameters

We use the analysis filterbank approach proposed in [12] to predetermine the basis set $\mathcal{P}$ and guide our choice of fixed parameters (priors, variances). The aim is to decompose the time series into seasonal components, and consider only those components which are highly coherent across the period, as well as have high energy, hence are important to modeling and prediction. In order to do this, we consider, at each time step, a single time period ending at the given time step and pass it through a filterbank. The resulting series of coefficients can then be analyzed.

Let $\boldsymbol{y}(t) = [y(t-p+1), y(t-p+2), \cdots, y(t)]^T$ be the data at hand. Then, from (2), using the 'complete' basis, we can write

$$\boldsymbol{a}^{\text{fil}}(t) = \boldsymbol{\Phi}^{-1}\boldsymbol{y}(t), \quad t = 1, \cdots, n, \tag{6}$$

where $a_j^{\text{fil}}(t)$ are the coefficients associated with the complete basis decomposition, $\boldsymbol{\Phi} = \left[\boldsymbol{\phi}_1, \cdots, \boldsymbol{\phi}_p\right]$ is the matrix of all the basis functions, and its inverse has an analysis filterbank interpretation. In other words, denoting the j-th row of $\boldsymbol{\Phi}^{-1}$ by $\boldsymbol{\Psi}_j^T = [\psi_j(p-1), \cdots, \psi_j(0)]$, (6) can be written as

$$a_j^{\text{fil}}(t) = \sum_{i=0}^{p-1} \psi_j(i) y(t-i), \quad j = 1, \cdots, p, \tag{7}$$

which is nothing but the output obtained on passing $y(t)$ through a filterbank consisting of p FIR filters; $[\psi_j(p-1), \cdots, \psi_j(0)]$ being the impulse response of the j-th filter. After having obtained the analysis filterbank output $a_j^{\text{fil}}(t)$ defined in (7), the data $y(t)$ can be reconstructed according to

$$y(\tau) = \sum_{j=1}^{p} a_j^{\text{fil}}(t)\phi_j(\tau), \quad \tau = t-p+1, \cdots, t, \tag{8}$$

which can be considered as the decomposition of y into p component waveforms, whose shapes are determined by the basis functions ϕ_j.

3.1.1. *Choice of Basis Set*

Clearly, with p being very large ($p = 288$ for our example), we aim to reduce the dimensions of the filterbank and chose $\mathcal{P}$ by analyzing $a_j^{\text{fil}}(t)$. In [6], two measures on the component waveforms are suggested to quantify the behavior of $a_j^{\text{fil}}(t)$ to aid in the selection of $\mathcal{P}$, namely, the *coherence* measure and the *energy* measure. The *coherence* measure is defined as

$$\hat{c}_j = \frac{\hat{\mu}_j^2}{\hat{\mu}_j^2 + \hat{\sigma}_j^2}, \tag{9}$$

where $\hat{\mu}_j^2$ is the sample mean and $\hat{\sigma}_j^2$ is the sample variance of $\left\{a_j^{\text{fil}}(t)\right\}_{t=1}^{n}$. We seek to include highly coherent waveforms (waveforms with high values of $\hat{c}_j$) in $\mathcal{P}$ as they have long lasting effects, making them good candidates for long-term forecasting. The *energy* measure of the component waveforms is defined as

$$\hat{E}_j = \frac{1}{n}\sum_{t=1}^{n}\left(a_j^{\text{fil}}(t)\right)^2 = \hat{\mu}_j^2 + \hat{\sigma}_j^2. \tag{10}$$

High energy components along with high coherence component are crucial to effective modeling of $y(t)$, and are included in $\mathcal{P}$. For future reference, let the number of basis functions included in $\mathcal{P}$ be K.

In this paper, we take sinusoids as the basis functions, and thus perform the short-term Fourier transform (STFT) on the weekday data $y(t)$ of our example. From the analysis of the STFT output, we note that only the fundamental frequency term ($\omega_0 = \frac{2\pi}{p}$), its first few harmonics, and the zero-frequency term have sufficiently high coherence as well as energy measure, while the rest of them appear to be insignificant in comparison. We select the first five frequencies (the fundamental frequency and its first four harmonics) together with DC (zero-frequency) to form $\mathcal{P}$. Since each frequency corresponds to two waveforms (a sine and a cosine), the dimension of $\mathcal{P}$ is $K = 11$. Thus we achieve our goal of reducing the dimensionality of the model, by bringing it down from $p = 288$ to 11.

3.1.2. Choice of Fixed Parameters

The initial state vector $\boldsymbol{a}(0)$, mean $\bar{\boldsymbol{a}}(0)$, and covariance matrix $\boldsymbol{P}_a(0)$ are computed as the respective mean and covariance of the output of the analysis filterbank on the training data.

For the dynamic selection of the basis set $\mathcal{P}_t$ in (5), it is assumed to follow a first order discrete Markov model given by $\Pr(\mathcal{P}_{t+1} = \rho_j | \mathcal{P}_t = \rho_i) \triangleq \pi_{ij}$, where the set ρ_i are some subsets of $\mathcal{P}$. The introduction of these transition probabilities offers flexibility in changing the harmonic regression order, thus allowing the algorithm to adaptively adjust according to the data.

The noise variances can be estimated by looking at the residuals. We choose a larger variance for the zero-frequency term as compared to the variance of the residual time series to be able to accommodate the outliners.

3.2. Online Estimation and Prediction by SMC

We track the model based on the set of available historical data using the sequential Monte Carlo (SMC) technique [7, 10]. Our aim is to obtain an online Monte Carlo approximation of the target distribution $p(\boldsymbol{a}(0:t), \mathcal{P}_{0:t} | y(1:t))$. With this goal, the SMC method keeps M sample streams $\left(\boldsymbol{a}^{(m)}(0:t), \mathcal{P}_{0:t}^{(m)}\right)$, together with the associated importance weight $\omega_t^{(m)}, m = 1 \cdots, M$, such that,

$$p(\boldsymbol{a}(0:t), \mathcal{P}_{0:t} | y(1:t)) \approx \sum_{m=1}^{M} \omega_t^{(m)} \delta[\boldsymbol{a}^{(m)}(0:t), \mathcal{P}_{0:t}^{(m)}], \tag{11}$$

where $\delta[\cdot]$ is a Dirac function (written with brackets instead of the conventional subscript to ease the reading).

We progress sequentially through each stream by extending at time t, the past particles $\left(\boldsymbol{a}^{(m)}(0:t-1), \mathcal{P}_{0:t-1}^{(m)}\right)$, by sampling $\left(\boldsymbol{a}^{(m)}(t), \mathcal{P}_t^{(m)}\right)$ according to a so-called trial distribution,

$$q\left(\boldsymbol{a}(t), \mathcal{P}_t | \boldsymbol{a}^{(m)}(0:t-1), \mathcal{P}_{t-1}^{(m)}, y(1:t)\right). \tag{12}$$

The importance weight $\omega_t^{(m)}$ associated with each stream can then be recursively updated as

$$\omega_t^{(m)} = \frac{p\left(\boldsymbol{a}^{(m)}(0:t), \mathcal{P}_{0:t}^{(m)} | y(1:t)\right)}{q\left(\boldsymbol{a}^{(m)}(0:t), \mathcal{P}_{0:t}^{(m)} | y(1:t)\right)} \propto \omega_{t-}^{(m)}$$
$$\times \frac{p\left(y(t) | \boldsymbol{a}^{(m)}(t), \mathcal{P}_t^{(m)}\right) p\left(\boldsymbol{a}^{(m)}(t), \mathcal{P}_t^{(m)} | \boldsymbol{a}^{(m)}(t-1), \mathcal{P}_{t-1}^{(m)}\right)}{q\left(\boldsymbol{a}^{(m)}(t), \mathcal{P}_t^{(m)} | \boldsymbol{a}^{(m)}(0:t-1), \mathcal{P}_{t-1}^{(m)}, y(1:t)\right)}. \tag{13}$$

The simplest choice for the trial distribution is to take the transition probability. With this choice, the weight update equation (13) reduces to $\omega_t^{(m)} = \omega_{t-1}^{(m)} \cdot p\left(y(t) | \boldsymbol{a}^{(m)}(t), \mathcal{P}_t^{(m)}\right)$.

The estimated long-term component at time t is then given by

$$\hat{y}_L(t|t) = \sum_{m=1}^{M} \left(\sum_{j \in \mathcal{P}_t^{(m)}} a_j^{(m)}(t) \phi_j(t) \right) \cdot \tilde{\omega}_t^{(m)}, \tag{14}$$

where $\tilde{\omega}_t^{(m)} = \frac{\omega_t^{(m)}}{\sum_{m=1}^{M} \omega_t^{(m)}}$ is the normalized importance weight corresponding to the m-th stream. The d-step predicted value using the long-term model is then given by

$$\hat{y}_L(t+d|t) = \sum_{m=1}^{M} \left(\sum_{j \in \mathcal{P}_t^{(m)}} a_j^{(m)}(t) \phi_j(t+d) \right) \cdot \tilde{\omega}_t^{(m)}. \tag{15}$$

3.3. Resampling-based Adaptive Shrinkage of the Basis Functions

The importance weights measure the quality of the Monte Carlo samples. As we proceed with the algorithm, the weights progressively get smaller and smaller, and after a while, only a few of the streams carry significant weights, while the rest of the samples become ineffective. To avoid this problem, resampling [7] is performed when the variance of the importance weights exceeds a certain predetermined threshold.

At the beginning of the SMC procedure, for each of the Monte Carlo sample, the sinusoids to be included are randomly drawn with probability proportional to their respective coherence values. At time $(t-1)$, let $\mathcal{P}_{t-1}^{(m)} \subseteq \mathcal{P}$ denote the set of sinusoids being used by the m-th sample stream. At time t, the set $\boldsymbol{S}$ of m samples can be divided into three subsets: $\boldsymbol{S}_0$, whose harmonic regression order is left unchanged, $\boldsymbol{S}_{+1}$, whose harmonic regression order is incremented by unity (up to a maximum of K), and $\boldsymbol{S}_{-1}$, whose order is decreased by unity (down to a minimum of 0). A particular sample finds place in the subsets $\boldsymbol{S}_{+1}, \boldsymbol{S}_{-1}$ and $\boldsymbol{S}_0$ with probabilities P_i, P_d, and $(1 - P_i - P_d)$ respectively. Thus, we obtain new set of basis functions $\mathcal{P}_t^{(m)} \subseteq \mathcal{P}$, associated with the m-th Monte Carlo stream, at time t. Following this step, the samples and the importance weights are updated using (12), and (13) respectively. We then check for the resampling condition, and if required, perform resampling.

4. SHORT-TERM MODELING AND PREDICTION

The long-term estimation error $z(t) = y(t) - \hat{y}_L(t|t)$ is employed as the raw data for the d-step prediction of the

short-term component, that covers both the short-term fluctuations and the long-term prediction error. We employ an autoregressive (AR) model, which is simple and effective in time series modeling for such data. Our short-term model can be cast into the following state-space model:

$$\begin{aligned} z(t) &= \sum_{i=1}^{q_t} b_i(t) \cdot z(t-d-i+1) + \varepsilon(t), \\ b_i(t) &= b_i(t-1) + w_i(t), \quad \forall i \in \mathcal{Q} \end{aligned} \tag{16}$$

where $\varepsilon(t)$ is the observation noise term, and $\boldsymbol{w}(t) = \{w_i(t), i \in \mathcal{Q}\}$ is the process noise. In order to model the bursts in the data, we use a heavy-tail distribution, such as a t-distribution, to model the observation noise density.

As was done in the long-term model case, the order q_t and the coefficients $\boldsymbol{b}(t)$ in the regression are tracked using SMC. To obtain an accurate prediction of the short-term model, here we also track the variance $\sigma_\varepsilon^2(t)$ of $\varepsilon(t)$. This is done as in [10], by modeling the evolution of the log-variance,

$$\log \sigma_\varepsilon^2(t) = \log \sigma_\varepsilon^2(t-1) + u(t), \tag{17}$$

where $\boldsymbol{w}(t)$ and $u(t)$ are zero-mean and have covariances $\boldsymbol{Q}_w$ and η respectively.

The sample streams are initialized by drawing samples of $\boldsymbol{b}(0)$ from a zero-mean Gaussian distribution with covariance $\boldsymbol{P}_b(0)$. Similarly, the initial set of samples for the noise parameter $\log \sigma_\varepsilon^2(0)$ is drawn from the Gaussian density with mean and variance $\mu_n(0)$, and $\sigma_n^2(0)$ respectively. The SMC algorithm here finds an estimate of the coefficients $\boldsymbol{b}(t)$ of the underlying AR process and the log-variance parameter $\log \sigma_\varepsilon^2(t)$ of the noise, based on the available short-term process $z(1:t)$. The target distribution $p\left(\boldsymbol{b}(0:t), q_{0:t}, \log \sigma_\varepsilon^2 | z(1:t)\right)$ can be factored as in Section 3.2 allowing for a recursive weight update.

Under this model, the d-step prediction of z based on the knowledge of $\{z(1), \cdots, z(t)\}$ is given by

$$\hat{z}(t+d|t) = \sum_{m=1}^{M_s} \tilde{\omega}_t^{(m)} \sum_{i=1}^{q_t^{(m)}} b_i^{(m)}(t|t) \cdot z(t-i+1), \tag{18}$$

where $\tilde{\omega}_t^{(m)}$ are the SMC weights similar to (15). Finally, the short-term prediction can be combined with the long-term prediction to obtain a complete d-step forecast as

$$\hat{y}(t+d|t) = \hat{y}_L(t+d|t) + \hat{z}(t+d|t). \tag{19}$$

Extending the idea of adaptive shrinkage of the harmonic regression discussed in Section 3.2, we select the order of the AR process modeling the short-term component adaptively via resampling, and also keep the provision of increasing or decreasing the order by introducing very small probabilities P_{in}, and P_{de}, which represent the probability of increasing and decreasing the order of the regression respectively. In other words, instead of keeping a fixed regression order, we let it evolve during the SMC procedure, and allow different streams to have different orders.

4.1. Computation of Confidence Bands

The SMC algorithm described above inherently provides a way of computing confidence bands since it carries information about the complete probability density function of the variables. Under the assumption that the entire randomness (error in the long-term prediction and remaining fluctuations) is carried by the short-term process $z(t)$, it is only necessary to find the confidence-band associated with this short-term process.

Let α denote the intended confidence level. We look for a α-confidence band which is symmetric and centered around the predicted value $\hat{z}(t+d|t)$. This can also be formulated as finding the smallest radius $\theta_\alpha(t)$ such that $[\hat{z}(t+d|t) - \theta_\alpha(t),\ \hat{z}(t+d|t) + \theta_\alpha(t)]$ contains a ratio α of the weights $\tilde{\omega}_t^{(m,n)} = \tilde{\omega}_t^{(m)}$ of the $M_s \cdot N_n$ samples. The final confidence band is then simply obtained by shifting the above confidence-level by the long-term prediction, giving $[\hat{z}(t+d|t) - \theta_\alpha(t),\ \hat{z}(t+d|t) + \theta_\alpha(t)]$.

5. NUMERICAL RESULTS

We use the example introduced in the beginning of this paper and present the performance of the proposed algorithm. We employ $M = 500$ Monte Carlo samples for both the long-term as well as short-term prediction. We cascade all the weekday data together to obtain the weekday regime.

Fig. 2 illustrates the performance of the algorithm for a 20-minute-ahead prediction horizon. It achieves a confidence level of 89.13%, as against an intended level of 90%, with RMSE equal to $.0712$. We also observed that the average regression order comes out to be approximately 4, while at the same time, we could actually have the regression order up to 8, allowing better modeling. Similarly, the average order of the harmonic regression comes out to be approximately 7.6, which is significantly below 11, and way below the original 288.

6. CONCLUSION

We have proposed a novel scheme for the forecasting of web-server workload time series which exhibits strong periodic patterns. A hierarchical framework is used to separately predict the long-term and the short-term components. The long-term forecast is performed using dynamic harmonic regression, while the residual short-term component is tracked as an autoregressive process. The coeffi-

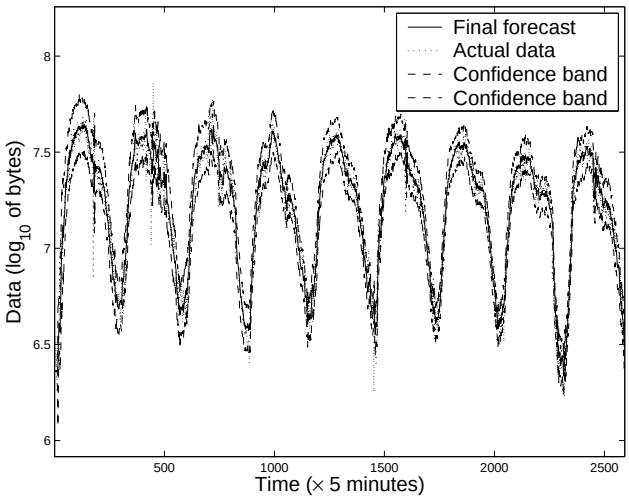

Fig. 2. 20-minute-ahead prediction for the weekday data, with 90% confidence band. RMSE of prediction is equal to 0.0712. Actual coverage of the confidence band is equal to 89.13%. Median width of the confidence band is equal to 0.212.

cients of both processes are tracked under a stochastic state-space setting. Also, the predictions yield simultaneous confidence bands, which can be used to support probability-based service-level agreements. Modeling the noise in the short-term model by a heavy-tailed distribution makes the algorithm robust to outliers in the data.

7. REFERENCES

[1] V.A.F. Almeida and D.A. Menasce, *Capacity Planning for Web Services: Metrics, Models, and Methods*, Prentice Hall, 2002.

[2] D.P. De Farias, A. King, M. Squillante, and B. Van Roy, "Dynamic control of web server farms," in *Proc. 2002 INFORMS Revenue Management Section Conference*, June 2002.

[3] J.R. Gallardo, D. Makrakis, and M. Angulo, "Dynamic resource management considering the real behavior of the aggregate traffic," *IEEE Trans. Multimedia*, vol. 3, no. 2, pp. 177–185, June 2001.

[4] M. Arlitt and C.L. Williamson, "Internet web servers: workload characterization and performance implications," *IEEE Trans. Networking*, vol. 5, no. 5, pp. 631–645, 1997.

[5] D. Shen and J.L. Hellerstein, "Predictive models for proactive network management: application to a production web server," in *Proc. 2000 IEEE/IFIP Network Operations and Management Symposium*, Apr. 2000, pp. 833–846.

[6] T.H. Li, "A hierarchical framework for modeling and forecasting web server workload," *J. Amer. Stat. Assoc.*, vol. 100, no. 471, pp. 748–763, Sep. 2005.

[7] R. Chen and J.S. Liu, "Sequential Monte Carlo methods for dynamic systems," *J. Amer. Stat. Assoc.*, vol. 93, pp. 1302–1044, 1998.

[8] X. Wang, R. Chen, and J.S. Liu, "Monte Carlo Bayesian signal processing for wireless communications," *J. VLSI Sig. Proc.*, vol. 30, no. 1-3, pp. 89–105, Jan.-Feb.-Mar. 2002.

[9] P.C. Young, D.J. Pedregal, and W. Tych, "Dynamic harmonic regression," *Journal of Forecasting*, vol. 18, pp. 369–394, 1999.

[10] C. Andrieu, M. Davy, and A. Doucet, "Efficient particle filtering for jump Markov systems - applications to time-varying autoregressions," *IEEE Trans. Sig. Proc.*, vol. 51, no. 7, pp. 1762–1770, 2003.

[11] D. Guo, X. Wang, and Rong Chen, "Wavelet-based sequential Monte Carlo blind receivers in fading channels with unknown channel statistics," *IEEE Trans. Sig. Proc.*, vol. 52, no. 1, pp. 227–239, Jan. 2004.

[12] T.H. Li and M.J. Hinich, "A filter bank approach for modeling and forecasting seasonal patterns," *Technometrics*, vol. 44, no. 1, pp. 1–14, 2002.